Amendments to the Specification:

Amend page 11 paragraph beginning at line 23 as follows:

Figure 5 discloses the computational model per the invention. This model, for computing N-bit at a time [500], works whichever generator polynomial G(X) is chosen provided a group under multiplication, as previously discussed, can be formed. G(X) is assumed to be of degree d, i.e., its higher term is Xd or, alternatively, it is represented under the form of a binary vector having d+1 terms. N is whatever value is necessary to be able to complete the job of generating a FCS [510], or checking a frame including a FCS [520] upon reception, in the imparted time. Then, frame is taken, from most significant bit [530], N-bit at a time [500]. Each chunk of N-bit must thus be reduced to the size of the FCS which corresponds to d, the degree of the generator polynomial. If N is greater than d then, as explained previously, the chunk of bits must be 'divided' by G(X) [535] until result is, at most, d-bit wide. In other words, one must add (i.e., subtract) G(X) as many times as necessary, from the N-bit vector, until its value is modulo G(X). Obviously, if N is already equal or lower than d then, nothing more but padding 0's on the left (if d<N) to match d, is to be done. After which current value of FCS [540], multiplied [560] by a displacement corresponding to N (xN_Multiplier), in the multiplicative group formed with G(X), must be added [550] so as to obtain the next current value. <u>This multiplication [560] is termed forward multiplication.</u> This loop is gone through as many times as necessary to check or generate FCS over a frame [520]. The result [570] is a d-bit wide vector result of the checking (an all zero's vector since remainder should be 0) or the FCS itself, to be padded to the transmitted frame. In practice, protocols always specify frame format so that enough room is obviously reserved to insert a FCS corresponding to G(X). Therefore, generation and checking are just about the same operation. At generation, FCS field is first blanked and result (computed with the blanks or 0's) is inserted into the reserved field before transmission. At checking, the frame to which the remainder of the division has thus been added, must return a zero value if everything has gone right en route. Again, this describes the basic process which may be somehow modified by protocols however, without bringing any fundamental change that could prevent the invention from being utilized though.

Amend page 15 paragraph beginning at line 1 as follows:

Figure 7 discusses another advantage of using the invention. If, for whatever reason, computation must start from FCS i.e., from the 'end' of the message or from the 'bottom' of a file so that FCS [710] is available first and checking must end with MSB [730] of frame [720] a simple modification to the scheme of the invention can be brought. It just consists in moving backward in the multiplicative group, by step of [N,]] N-bit at a time [700] thus requiring the use of a x^{-} N Multiplier [760] instead of the forward xN Multiplier of figure 5. This is the only change which is necessary to accommodate a backward checking. This multiplication [760] is termed backward multiplication.

Amend page 17 paragraph beginning at line 1 as follows:

Figure 11 shows the steps of the method for computing, N-bit at a time, according to the invention. Computation starts or resume at step [1100] when a first or a subsequent chunk of N-bit is picked from the piece of data e.g., a frame, to be checked or encoded before transmission. Then, the N-bit wide chunk is divided modulo G(X) ~~[1100]~~ [1110] so as result matches the degree (d) of the generator polynomial. An actual division need to be performed only if N is larger than d. If equal, nothing is to be done i.e., the N-bit wide chunk of data is used as is. If lower, the N-bit chunk of data, used as is, must also be left justified i.e., padded with enough zeros to match the size of the d-wide adder. The above is not different from the division of regular integers where if dividend is already equal or lower than divider then, nothing specific is to be done. While performing division (or sequentially) the current value of FCS, considered as a vector of the multiplicative group that can be formed with G(X), is multiplied as it is displaced by a value corresponding to N [1120] in the multiplicative group. Both are added [1130] and result used to update FCS [1140] which becomes the current value. If more data are to be processed [1151] method resumes at step [1100] with N more bits entering the state machine. If not, loop is exited [1152]. Result of the checking or the pattern of bits to update FCS field is therefore available in FCS register.

Amend page 18 paragraph beginning at line 3 as follows:

Figure 13 shows the steps of the method for generating automatically the structure of the divider and multiplier needed to perform a backward computation (from LSB) of CRC's. The first part is identical to figure 12. That is, computation of vectors start from a0 [1300]. N 'positive' vectors are generated by successive multiplication's [1320] by a1. All vectors are recorded [1310]. When enough vectors (N) [1330] for the divider have been obtained [1332], (identical to the ones of figure 12) those for the x^{-} N Multiplier are generated in turn from a-1 after the last vector of the group has been deduced from G(X) [1340] so as, by successive multiplication of a-1 [1360], restarting from a0 [1350], N 'negative' vectors are generated [1380] and recorded ~~[1360]~~ [1370]. When done [1382], like in figure 12, vectors are used [1390] for synthesizing the corresponding logic of XOR's especially, implementing in this case, a backward FCS multiplier in place of the forward multiplier of figure 12.

Amend page 20, add New paragraphs after line 7 as follows:

In summary, a general way of performing a Cyclic Redundancy Check (CRC) calculation, N-bit at a time, is disclosed. CRC calculation is based on a generator polynomial G(X) of degree d so that all results always fit a d-bit wide Field Check Sequence (FCS). The generator polynomial allows forming a multiplicative cyclic group comprised of d-bit wide binary vectors. The iterative calculation method assumes that each new N-bit chunk of data bits, picked from the binary string of data bits, is divided, modulo the generator polynomial G(X), so that to obtain a d-bit wide division result while a current value of the d-bit wide FCS is displaced in the multiplicative cyclic group, of a value corresponding to N. Then, the d-bit wide division result and the displaced d-bit wide FCS are added to become the new current FCS. The above steps are re-executed until no data bits are left thus, getting the final result of the CRC calculation which can be used either for checking or generation of FCS. The method of the invention allows a forward (from MSB - Most Significant Bit) or backward (from LSB - Least Significant Bit) calculation of CRC's. The invention also provides for an automatic generation of the logic necessary to actually carry out the calculations thus, does not assume any particular skill on CRC's when used. It applies, irrespective of the degree of the generator polynomial in use (d) and whatever value of N is picked.

Although the present invention has been described with reference to preferred embodiments artisans skilled in the art will recognize that changes may be made in form and detail without departing from the spirit and scope of the invention.